MC24 Software Manual

Release 6.12/1.0

Embention Sistemas Inteligentes, S.A.

2025-02-24

Contents

Software applications 3
Veronte Link 3
MC24 PDI Builder 3
List of Variables 4
BIT Variables 4
Real Variables (RVar) - 32 bits 6
Integer Variables (UVar) - 16 bits 11
CAN Bus protocol 13
CAN Commands to MC24 13
Telemetry messages from MC24 14

Software applications

To properly connect and configure a **MC24 unit**, **Veronte Link** and **MC24 PDI Builder** are required.

Veronte Link

Veronte Link establishes communication between a computer and any Veronte product by creating a VCP bridge. It allows to use multiple control stations and autopilots to be interconnected, operating simultaneously.
Veronte Link also includes a post-flight viewer, to reproduce all recorded data from previous flights and generate plots and reports.

For more information, visit the Veronte Link user manual.

MC24 PDI Builder

This tool is used to set all the configurable parameters. Here the user can set, tune and define the motor, control systems and sensors that are going to be used alongside the ESC.

For more information, visit the MC24 PDI Builder user manual.

List of Variables

This section shows the variables employed exclusively by **Veronte MC24**. The rest of variables can be read in the Lists of Variables - Lists of interest section of **1x Software Manual**.

BIT Variables

ID	Name	Description
400	C1 Low Frequency	Low priority task frequency - Dependent on CIO Running Frequency (RVar 2057) • 0 for error → CIO Running Frequency < 10 Hz • 1 for OK → CIO Running Frequency > 10 Hz
402	Acquisition Step Missed	• 0 for Acquisition step missed → High priority task frequency fluctuation is higher than permitted (1%). • for Acquisition Task OK → High priority task frequency fluctuation is under set limits (1%).
482	MC Hall Sensor error	0 for error, 1 for OK
483		0 for error, 1 for OK

ID	Name	Description
	MC Sin/Cos Sensor error	
484	MC General health error	0 for health error, 1 for status OK
485	MC Current sensing error	0 for error, 1 for OK
486	MC Phase U Current Calibration Error	ADC phase U not calibrated - 0 for not calibrated, 1 for calibration OK
487	MC Phase V Current Calibration Error	ADC phase V not calibrated - 0 for not calibrated, 1 for calibration OK
488	MC Phase W Current Calibration Error	ADC phase W not calibrated - 0 for not calibrated, 1 for calibration OK
489	MC Maximum Temperature Error	Maximum power module temperature exceeded - 0 for error (exceeded), 1 for OK
490	MC Phase Error	Power module driver phase error reported - 0 for error, 1 for OK
491	MC General Driver Error	Power module driver error reported - 0 for error, 1 for OK
492	MC Over-current AC	Current AC side limit exceeded - 0 for error (exceeded), 1 for OK
493		

ID	Name	Description
	MC Over-voltage advertisement	Over-voltage DC side limit advertisement exceeded - 0 for error (exceeded), 1 for OK
494	MC Over-voltage caution	Over-voltage DC side limit caution exceeded - 0 for error (exceeded), 1 for OK
495	MC Under-votage latching	Critical under-voltage DC side limit violation - 0 for error, 1 for OK
496	MC Under-votage non latching	Non critical under-voltage DC side limit violation - 0 for error, 1 for OK
497	MC RMS imbalance	Current AC side imbalance - 0 for error, 1 for OK
498	MC Open DC fault	Open-circuite DC side fault - 0 for error, 1 for OK
499	MC Over-current DC	Current DC side limit exceeded - 0 for error (exceeded), 1 for OK

Real Variables (RVar) - 32 bits

ID	Name	Units/ Values	Description
2057	CIO Running Frequency	Hz	Low priority task running frequency
2058		Hz	

ID	Name	Units/ Values	Description
	CIO Min Running Frequency		Minimum assured frequency of low priority task
2330	Control Loop Period	s	MC control loop period
2331	Control Loop Maximum Period	s	MC maximum control loop period
2332	Control Loop Duration	s	MC control loop average execution time
2333	MC Control Loop Maximum Duration	s	MC control loop maximum execution time
2335	MC Control Loop Maximum CPU Usage Ratio	%	MC maximum CPU usage ratio
2336-2338	MC U-V-W Phase Current	A	MC U-V-W phase current
2339	MC Electrical Angle	rad	MC electrical angle
2340	MC Mechanical Angle	rad	MC mechanical angle
2341	MC Mechanical Angular Speed	rad/s	MC mechanical angular speed

ID	Name	Units/ Values	Description
2342	MC Desired Mechanical Angle	rad	MC desired mechanical angle
2343	MC Position Controller Output	rad/s	MC position PDI output
2344	MC Desired Mechanical Angular Speed	rad/s	MC desired mechanical angular speed
2345	MC Desired Mechanical Angular Speed After Speed Limiter	rad/s	MC desired mechanical angular speed after speed limiter
2346	MC Speed Controller Output	A	MC speed PDI output
2347-2348	MC Clarke Alpha-Beta Current	A	MC alpha and beta current after Clarke transformation
2349-2350	MC Actual Direct-Quadrature Current	A	MC actual direct-quadrature currents
2351-2352	MC Desired Direct-Quadrature Current	A	MC desired direct-quadrature currents
2353-2354		V	MC current PIDs outputs

ID	Name	Units/ Values	Description
	MC Direct-Quadrature Voltage From Controller Output		
2355-2356	MC Alpha-Beta Voltage From Current Controller Output	V	MC Clarke alpha-beta currents
2362-2364	MC U-V-W Phase PWM Duty Cycle	%	MC PWM outputs
2367	MC Mechanical Angle Error	rad	MC mechanical angle error
2368-2370	MC U-V-W Phase BEMF	V	MC U-V-W phase electromechanical force
2371	MC Input Current DC side	A	DC bus current
2372	MC Input Normalized Command Speed	customType	Speed input rate from source (CAN or PWM)
2373-2374	MC ADC in 0-1	V	**Note** System reserved variables

ID	Name	Units/ Values	Description
2375	MC Logic Board Temperature	K	Board temperature
2376	MC Power Module Temperature	K	IGBT filtered temperature
2377	MC Motor Temperature	K	Motor temperature
2378	MC Input Voltage DC side	V	DC bus voltage
2379-2380	MC U-V Phase Hall current senso	customType	**Note** System reserved variables
2381	MC Virtual and estimator angle difference	rad	Angle offset value from estimated and commanded angle to close control loop
2382	MC Low speed estimator angle	rad	Low speed observer estimated angle
2383	MC High speed estimator angle	rad	High speed observer estimated angle
2384		rad/s	

ID	Name	Units/ Values	Description
	MC Low speed estimator speed		Low speed observer estimated mechanical speed
2385	MC High speed estimator speed	rad/s	High speed observer estimated mechanical speed

Integer Variables (UVar) - 16 bits

ID	Name	Description
801	Control Mode	Index of the control input mode: • 1: PPM • 2: CAN • 3: Both modes active (CAN priority)
802	Actual State	State of motor controller: • 1: Motor stop and driver disabled • 2: Calibration of ADC reading

ID	Name	Description
		• 3: Initial alignment procedure • 4: Open loop procedure • 5: Speed mode

CAN Bus protocol

CAN Commands to MC24

MC24 can receive commands from any CAN device. All CAN messages for **MC24** follow the same structure, a string of bits divided in two groups:

Group	Name	Size	Description
1	CAN Id	11-bits: standard 29-bits: extended	If the **CAN Id** matches with the Id of a **MC24** input filter (a type of CAN consumer), the message will be correctly read by the MC speed filter consumer. Otherwise, it will be ignored.
2	Payload	4 bytes	**Speed** must be represented with a **compressed 32-bit signed** variable with little endian format. The values of this variable should be in the range [**0 to maximum RPM** (speed)] or [**-maximum RPM to maximum RPM**], depending on whether users want to allow negative commands. **Note** Negative values command the opposite movement to positive ones. Therefore, the maximum value corresponds with the maximum speed and the maximum negative value

Group	Name	Size	Description
			corresponds with the maximum reversed speed.

The parameter that is configured in the **MC24** to receive these CAN commads is the **CAN Id** of **Input filter** producer, which has to be linked to the **CAN Cmd** consumer. To know more, read the CAN I/O - Input/Output section of the **MC24 PDI Builder** user manual.

An example for sending commands from **Veronte Autopilot 1x** to a **MC24** unit is explained in the MC110/MC24 - Integration examples section of the **1x PDI Builder** user manual.

Telemetry messages from MC24

Telemetry messages can be transmitted from the **MC24** unit to provide information of interest to the user, such as the board temperature or the input command values.

CAN messages sent by **MC24** have also the structure:

1. **CAN Id**: It can be in standard frame format (11-bits) or in extended frame format (29-bits). The CAN Id frame format will depend on the CAN protocol of the receiving device.
2. **Variable**: Users can send as telemetry the variable they want to know information about. All the variables available to be sent from the **MC24** unit can be consulted in the Lists of Variables section of this manual.
   The format in which these variables must be sent will depend on the CAN protocol of the device that will read the message.

Detailed information on **how to build CAN messages** can be consulted in the Custom Messages types - Input/Output section of the **1x PDI Builder** user manual.